Implementasi *Block Based Watermarking* pada Citra *Digital*

Artikel Ilmiah

Peneliti:
Yuliana Onna Bebut (672010068)
Magdalena A. Ineke Pakereng, M.Kom.

Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
Agustus 2016

Implementasi *Block Based Watermarking* pada Citra *Digital*

Artikel Ilmiah

Diajukan kepada
Fakultas Teknologi Informasi
untuk memperoleh gelar Sarjana Komputer

Peneliti:
Yuliana Onna Bebut (672010068)
Magdalena A. Ineke Pakereng, M.Kom.

Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
Agustus 2016

PERPUSTAKAAN UNIVERSITAS
UNIVERSITAS KRISTEN SATYA WACANA
Jl. Diponegoro 52 – 60 Salatiga 50711
Jawa Tengah, Indonesia
Telp. 0298 – 321212, Fax. 0298 321433
Email: library@adm.uksw.edu ; http://library.uksw.edu

PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama	: Yulianaa Onna Bebut		
NIM	: 672010068	Email	: sonna_yuli@yahoo.com
Fakultas	: Teknologi Informasi	Program Studi	: Teknik Informatika

Judul tugas akhir : Implementasi *Block Based Watermaking* pada Citra *Digital*

Pembimbing : Magdalena A. Ineke Pakereng, M.Kom.

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar kesarjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 22 September 2016

Yuliana Onna Bebut

PERPUSTAKAAN UNIVERSITAS
UNIVERSITAS KRISTEN SATYA WACANA
Jl. Diponegoro 52 – 60 Salatiga 50711
Jawa Tengah, Indonesia
Telp. 0298 – 321212, Fax. 0298 321433
Email: library@adm.uksw.edu ; http://library.uksw.edu

PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama	: Yuliana Onna Bebut		
NIM	: 672010068	Email	: sonna_yuli@yahoo.com
Fakultas	: Teknologi Informasi	Program Studi	: Teknik Informatika
Judul tugas akhir	: Implemtasi *Block Based Watermarking* Pada Citra *Digital*		

Dengan ini saya menyerahkan hak *non-eksklusif** kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

☑ a. Saya mengijinkan karya tersebut diunggah ke dalam aplikasi Repositori PerpustakaanUniversitas, dan/atau portal GARUDA

☐ b. Saya tidak mengijinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA**

* *Hak yang tidak terbatashanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-ekslusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.*

** *Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).*

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 22 September 2016

Yuliana Onna Bebut

Mengetahui,

Magdalena A. Ineke Pakereng, M.Kom.
Pembimbing

FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
Jalan Diponegoro 52 – 60
Phone. (0298) 321212 (Hunting)
Fax. (0298) 321433
E-mail: fti@uksw.edu
Salatiga 50711 – INDONESIA

LEMBAR PERSETUJUAN PUBLISH JURNAL

Dengan mempertimbangkan isi dari jurnal mahasiswa :

Nama Mahasiswa : Yuliana Onna Bebat
NIM : 672010068

Maka jurnal ini dinyatakan :

LAYAK TERBIT / ~~TIDAK LAYAK TERBIT~~

Menyetujui,

Magdalena A. Lucky Pakuny

Pembimbing 1

Pembimbing 2

Mengetahui,

Dr. Irwan Sembiring, ST, M.Com

Reviewer

Lembar Pengesahan

Judul Tugas Akhir	: Implementasi *Block Based Watermarking* Pada Citra *Digital*
Nama Mahasiswa	: Yuliana Onna Bebut
NIM	: 672010068
Program Studi	: Teknik Informatika
Fakultas	: Teknologi Informasi

Menyetujui,

Magdalena A. Ineke Pakereng, M.Kom.
Pembimbing

Mengesahkan,

Dr. Dharmaputra T. Palekahelu, M.Pd.
Dekan

Dr. Dharmaputra T. Palekahelu, M.Pd.
Pjs. Ketua Program Studi

Dinyatakan Lulus Tanggal: 9 September 2016

Reviewer :

- Dr. Irwan Sembiring, ST., M.Kom.

Implementasi *Block Based Watermarking* pada Citra *Digital*

[1] Yuliana Onna Bebut, [2]M. A. Ineke Pakereng
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
Email: [1]672010068@student.uksw.edu,[2]inekep200472@yahoo.com

Abstract

Documents in the form of digital image has the possibility to be manipulated unlawfully. The information contained within can be faked, so that the recipient of the document can be wrong in interpreting the intention of the information therein. This could result in losses for both the sender and recipient document document, the decision made, based on the information that has been falsified. A solution is needed to secure the information stored on it. Information from the sender must be the same when it reached the receiver. In this study generated watermarking applications implemented in a way that is calculating the value of bytes of data into a form MD5 hash algorithm. Results hash is then encrypted with an algorithm Vigenere. Cipher hash then inserted at Least Significant Bit of digital images. Watermarking is inserted can be used to detect whether the digital image has been changed or not. The test results showed that the change can be detected, even if the change only by 2x2 pixels.

Keywords: *Watermarking, Least Significant Bit Embedding, Vigenere*

Abstrak

Dokumen berbentuk citra *digital* memiliki kemungkinan untuk dimanipulasi secara tidak sah. Informasi yang terdapat di dalamnya dapat dipalsukan sehingga pihak penerima dokumen dapat salah dalam menginterpretasikan maksud informasi di dalamnya. Hal ini dapat mengakibatkan kerugian baik bagi pengirim dokumen maupun penerima dokumen, karena keputusan yang dibuat, berdasarkan pada informasi yang telah dipalsukan. Sebuah solusi diperlukan untuk mengamankan informasi yang tersimpan di dalamnya. Informasi dari pengirim harus sama ketika sampai di penerima. Pada penelitian ini dihasilkan aplikasi *watermarking* diimplementasikan dengan cara yaitu menghitung nilai *byte* data menjadi bentuk *hash* dengan algoritma MD5. Perhitungan dilakukan per blok warna pada citra *digital* sebanyak 128 warna. Hasil *hash* sepanjang 128 *bit* kemudian dienkripsi dengan algoritma *Vigenere*. *Cipher hash* kemudian disisipkan pada bagian *Least Significant Bit* citra *digital*. *Watermarking* yang disisipkan tersebut dapat berfungsi untuk mendeteksi apakah citra *digital* telah mengalami perubahan atau tidak. Hasil pengujian menunjukkan bahwa perubahan dapat terdeteksi, sekalipun perubahan hanya sebesar 2x2 piksel.

Kata Kunci: Tanda Tangan *Digital*, Penyisipan *Least Significant Bit, Vigenere*

[1]Mahasiswa Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana
[2]Staf Pengajar Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

1. Pendahuluan

Pada masa kini, komputer dan internet sudah menjadi kebutuhan utama. Hampir semua orang menggunakan komputer maupun internet dalam kehidupan mereka sehari-hari, baik untuk keperluan pendidikan, bisnis, hiburan, dan lain-lain. Namun, seiring dengan pesatnya perkembangan teknologi informasi khususnya dalam hal yang berkaitan dengan komputer dan internet, masalah keamanan data juga semakin kompleks.

Beberapa masalah keamanan tersebut adalah pencurian serta pemalsuan data dan dokumen cetak maupun *digital*. Data-data yang terdistribusi dalam internet dan database dapat dengan mudah dimanipulasi oleh pihak yang tidak bertanggung jawab. Salah satu cara untuk mencegahnya adalah dengan membuat suatu tanda khusus yang memastikan bahwa data tersebut adalah data benar dan otentik serta mempunyai syarat integritas data. Untuk itu dapat digunakan salah satu teknologi keamanan data yang disebut dengan *watermarking*.

Watermarking atau yang juga disebut tanda tangan *digital* adalah suatu mekanisme untuk menggantikan tanda tangan secara manual pada dokumen kertas [1]. *Watermarking* memiliki fungsi sebagai penanda pada data yang memastikan bahwa data tersebut adalah data yang sebenarnya. Penanda pada *watermarking* ini tidak semata hanya berupa tanda tangan *digital*, tetapi dapat berupa cap *digital*, *text*, *bit*, dan gambar. Aspek keamanan dan kerahasiaan bukan disediakan dengan sistem berupa tanda tangan *digital*, tetapi tanda tangan yang telah dienkripsi terlebih dahulu dengan algoritma tertentu. *Watermarking* pada citra *digital*, pada umumnya meringkas keseluruhan citra *digital* menjadi sebuah nilai *hash* dengan panjang sesuai algoritma yang digunakan. Jika menggunakan MD5, maka dihasilkan *hash* dengan panjang 128 *bit* [1], jika menggunakan CRC-16, maka dihasilkan *hash* dengan panjang 17 *bit* [2] dan lain sebagainya. Pada *block based watermarking*, *hash* dihitung bukan berdasarkan keseluruhan file citra *digital*, namun terbagi ke dalam blok-blok. Tiap blok menyimpan *bit-bit hash*. Keuntungan dari *block based watermarking* adalah *watermark* di dalamnya lebih sensitif terhadap manipulasi.

Berdasarkan latar belakang tersebut maka dilakukan penelitian dengan judul "Implementasi *Block Based Watermarking* pada Citra *Digital* ", yang diharapkan dapat membantu menyedikan alat untuk menjaga integritas informasi dari suatu citra *digital*.

2. Tinjauan Pustaka

Pada penelitian yang berjudul "*Perancangan Aplikasi Watermarking Pada Media Fotografi Sebagai Perlindungan Hak Cipta Menggunakan Metode Spread Spectrum*", yang membahas tentang teknik penyembunyian data atau informasi yang bersifat rahasia ke data lain. Informasi disandikan dengan menggunakan *Spread Spectrum*. Pada penelitian dihasilkan aplikasi yang dapat menyisipkan pesan (*watermark*) pada citra digital. Watermark dimasukkan oleh pengguna aplikasi [3].

Penelitian lainnya yang berjudul "Teknik Pembuktian Kepemilikan Citra Digital Dengan *Watermaking* Pada *Domain Wavelet*" Dalam penelitiannya di bahas mengenai *Watermarking* merupakan salah satu solusi dalam memecahkan permasalahan pembuktian kepemilikan data digital. Pada penelitian ini teknik *watermarking* pada citra digital menggunakan *wavelet* sebagai media *transformasinya* (DWT).Citra *original* ditransformasi menggunakan *wavelet* menjadi empat area *frekuensi* LL, LH, HL, dan HH.*Bit-bit watermark* ditanam pada area LH dan HL. Kualitas citra ter-*watermark* diamati berdasarkan nilai *Peak Signal of Noise Ratio* (PSNR) [4].

Berdasarkan penelitian-penelitian dilakukan tentang pemanfaatan *watermarking* untuk mengamankan citra digital, maka dilakukan penelitian yang membahas mengenai perancangan *block based watermarking*. Penelitian ini menggunakan algoritma MD5 untuk mendapatkan nilai *hash* yang nantinya akan dienkripsi dan disisipkan dalam citra *digital*. Perbedaan penelitian ini dengan penelitian sebelumnya adalah, (1) pesan yang disisipkan sebagai watermark, bukan merupakan pesan yang dimasukkan oleh pengguna namun berupa hasil *output* proses *hashing* dengan algorimta MD5. *Input* untuk algoritma MD5 bukan berdasarkan keseluruhan *byte* citra *digital*, namun dikelompokkan per blok dengan ukuran 128 *byte*. Angka 128 *byte* digunakan karena menyesuaikan dengan ukuran blok output dari MD5. Proses ini akan dijelaskan lebih lanjut pada bagian perancangan; (2) pesan yang berupa nilai hash, dienkripsi dengan algoritma Vigenere menghasilkan cipher hash; (3) cipher *hash* tersebut, kemudian disisipkan pada LSB masing-masing blok tersebut. Hal ini memberikan kelebihan, yaitu dapat dilakukan lokasi kerusakan citra *digital* karena *watermark* disisipkan tiap 128 *byte* warna (1 *byte* menyimpan 1 *bit watermark*).

Penelitian yang dilakukan dibangun dengan menggunakan teknik *Watermarking*. Istilah *watermarking* ini muncul dari salah satu cabang ilmu yang disebut dengan steganografi. Steganografi merupakan suatu cabang ilmu yang mempelajari tentang bagaimana menyembunyikan suatu informasi "rahasia" di dalam suatu informasi lainnya. Steganografi mempunyai sejarah yang hampir sama dengan kriptografi, keduanya banyak digunakan terutama pada zaman perang. *Watermarking* diartikan sebagai suatu teknik penyembunyian data atau informasi "rahasia" kedalam suatu data lainnya untuk "ditumpangi" (kadang disebut dengan *host* data), tetapi orang lain tidak menyadari kehadiran adanya data tambahan pada *host*. Jadi seolah-olah tidak ada perbedaan antara data *host* sebelum dan sesudah proses *watermarking*. Perbedaan watermarking dengan steganografi terletak pada fungsi pesan yang disisipkan. Pada steganografi, pesan disisipkan dengan tujuan untuk disembunyikan sehingga tidak diketahui oleh orang lain. Pada watermarking, pesan disisipkan dengan tujuan untuk melindungi media yang disisipi. Perlindungan ini dapat dalam bentuk bukti kepemilikan, atau bukti bahwa media masih utuh [5].

Menurut Amin [6], cara paling umum menyisipkan pesan adalah dengan memanfaatkan *Least Significant Bit* (*LSB*). Walaupun ada kekurangan pada metode ini, tetapi kemudahan implementasinya membuat metode ini tetap digunakan sampai sekarang. Metode ini membutuhkan syarat, yaitu jika dilakukan kompresi pada *image watermarking*, harus digunakan format *lossless*

compression, karena metode ini menggunakan beberapa *bit* pada setiap p*ixel* pada *image*. Jika digunakan format *lossy compression*, pesan rahasia yang disembunyikan dapat hilang. Jika digunakan image 24 *bit color* sebagai *cover*, sebuah *bit* dari masing-masing komponen *Red, Green* dan *Blue* dapat digunakan sehingga 3 *bit* dapat disimpan pada setiap *pixel*.

Pada susunan *bit* di dalam sebuah *byte* (1 *byte* = 8 *bit*), ada *bit* yang paling berarti *(most significant bit* (MSB) dan *bit* yang paling kurang berarti *least significant bit* (LSB). *Bit* yang cocok untuk diganti adalah *bit LSB*, sebab perubahan tersebut hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya. Misalkan pada *cover* citra, *byte* tersebut menyatakan warna merah, maka perubahan satu *bit* LSB tidak mengubah warna merah tersebut secara berarti, apalagi mata manusia tidak dapat membedakan perubahan yang kecil.

Misalnya, di bawah ini terdapat 3 piksel dari *image* 24 *bit color* :

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

Yang ingin disisipkan adalah huruf A dengan biner 01000001, dengan menyisipkannya ke dalam piksel di atas maka akan dihasilkan

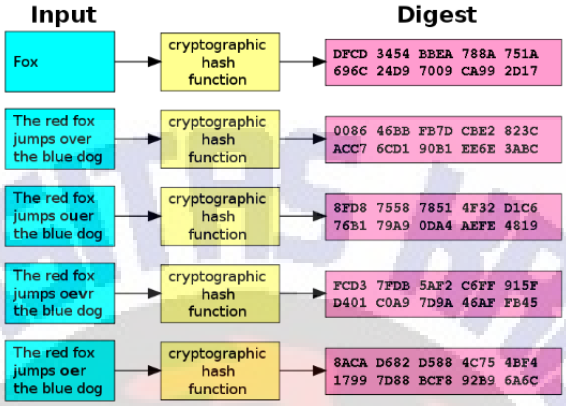
(00100111 11101001 11001000)
(00100111 11001000 11101000)
(11001000 00100111 11101001)

Dapat dilihat bahwa hanya 3 *bit* saja yang perlu diubah untuk menyembunyikan karakter A ini. Perubahan pada LSB ini akan terlalu kecil untuk terdeteksi oleh mata manusia sehingga pesan dapat disembunyikan secara efektif. Jika digunakan *image* 8 *bit color* sebagai *cover*, hanya 1 *bit* saja dari setiap piksel warna yang dapat dimodifikasi sehingga pemilihan *image* harus dilakukan dengan sangat hati-hati, karena perubahan LSB dapat menyebabkan terjadinya perubahan warna yang ditampilkan pada citra.

Teknik *block-based watermarking* bekerja dengan cara membagi media ke dalam blok-blok. Nama *block-based* ini tidak ada hubungan dengan *block* cipher pada kriptografi. *Watermark* pada citra *digital*, pada umumnya meringkas keseluruhan citra *digital* menjadi sebuah nilai *hash* dengan panjang sesuai algoritma yang digunakan. Fungsi *hash* dapat menggunakan algoritma-algoritma *hash* yang sudah ada, seperti MD5, SHA, atau CRC [7]. Jika menggunakan MD5, maka dihasilkan *hash* dengan panjang 128 *bit* [1], jika menggunakan CRC-16, maka dihasilkan *hash* dengan panjang 17 *bit* [2] dan lain sebagainya. *Block based watermark* menggunakan proses penyisipan yang dilakukan per-blok. *Bit* yang disisipkan adalah hasil operasi *hash* dari blok media. Pada *block based digital signature*, *hash* dihitung bukan berdasarkan keseluruhan file citra *digital*, namun terbagi ke dalam blok-blok. Tiap blok menyimpan *bit-bit hash*. Keuntungan dari *block based digital signature* adalah *watermark* di dalamnya lebih sensitif terhadap manipulasi. Keuntungan yang lain adalah, perubahan pada citra *digital*, dapat terdeteksi sampai pada level blok yang digunakan.

MD5 (*Message-Digest Algorithm* 5) ialah fungsi *hash* kriptografi yang digunakan secara luas dengan *hashvalue* 128-*bit*[8]. MD5 telah dimanfaatkan

secara bermacam-macam pada aplikasi keamanan, dan MD5 juga umum digunakan untuk melakukan pengujian integritas (*fingerprint*) sebuah *file*.MD5 didesain oleh Ronald Rivest pada tahun 1991 untuk menggantikan *hashfunction* sebelumnya, yaitu MD4.

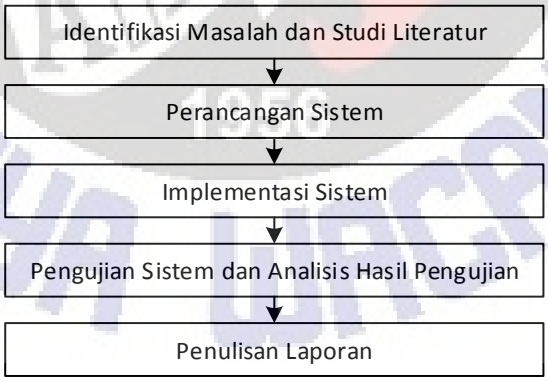


Gambar 2 *Hash Value* dari Beberapa *Input* yang Berbeda [1]

Hashvalue yang dihasilkan oleh MD5 memiliki panjang 128-*bit* (16 *byte*), sekalipun *input* (pesan) yang digunakan memiliki panjang yang bervariasi. *Hashvalue* berubah signifikan sekalipun perubahan yang terjadi pada *input* hanya 1 *byte* (1 kata).

3. Metode dan Perancangan Sistem

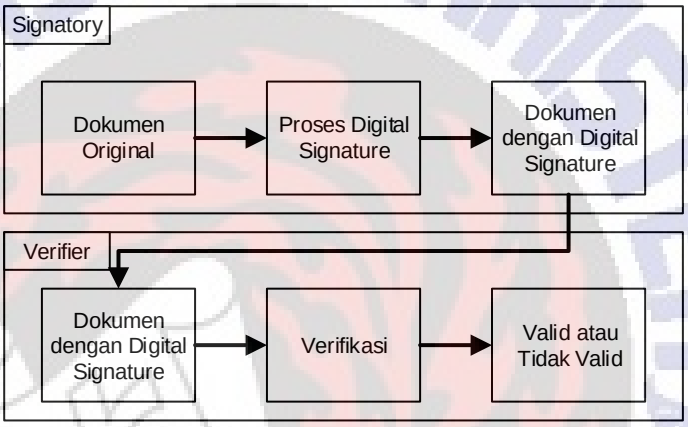
Penelitian yang dilakukan, diselesaikan melalui tahapan penelitian yang terbagi dalam lima tahapan, yaitu: (1) Identifikasi masalah dan studi literatur, (2) Perancangan sistem, (3) Implementasi sistem, (4) Pengujian sistem dan analisis hasil pengujian, (5) Penulisan laporan.



Gambar 3 Tahapan Penelitian

Tahapan penelitian pada Gambar 3, dapat dijelaskan sebagai berikut. *Tahap pertama:* Pada tahap ini dilakukan identifikasi masalah diperlukannya sistem pengamanan dokumen *digital* terutama citra *digital*. Selain identifikasi masalah, dikumpulkan juga penelitian-penelitian terdahulu yang membahas masalah yang sama atau mirip, sehingga dapat dilihat metode-metode yang dapat diaplikasikan untuk pengamanan citra *digital*. *Tahap kedua*: yaitu melakukan

perancangan sistem yang meliputi perancangan proses. Proses terbagi pada 2 bagian utama yaitu proses pemberian *watermark*, dan proses verifikasi *watermark*. Pada kedua proses tersebut, masing-masing terdapat subproses, yaitu proses pembuatan *hash*, proses enkripsi/dekripsi *hash*, dan proses *embedding/extracting watermark*. *Tahap ketiga*: yaitu mengimplementasikan rancangan yang telah dibuat di tahap dua ke dalam sebuah aplikasi/program sesuai kebutuhan sistem. *Tahap keempat*: yaitu melakukan pengujian terhadap sistem yang telah dibuat, serta menganalisis hasil pengujian tersebut, untuk melihat apakah aplikasi yang telah dibuat sudah sesuai dengan yang diharapkan atau tidak, jika belum sesuai maka dilakukan perbaikan. *Tahap kelima:* melakukan penulisan laporan penelitian.



Gambar 4 Arsitektur Sistem

Arsitektur sistem ditunjukkan pada Gambar 3. Sistem terdiri dari dua proses, yaitu proses *signing* dan proses *verifiying*. Proses *signing* dilakukan oleh *signatory*, dengan menanamkan *watermark* ke dalam dokumen. Proses *verifiying* dilakukan oleh *verifier*, dengan mengekstraksi *watermark* dari dalam dokumen.

1	2	3
4	5	6
7	8	9
10	11	12

Gambar 5 Contoh Pembagian Blok

Pada Gambar 5 diberikan contoh citra *digital* dengan ukuran 12 x 4 piksel. Pada contoh tersebut, dimisalkan bahwa blok yang digunakan berukuran 4 piksel (atau 12 warna, 1 piksel 3 warna). Tiap blok dihitung nilai *hash*, kemudian nilai *hash* tersebut disisipkan pada LSB blok itu sendiri. Aturan yang perlu diperhatikan adalah:

`Jumlah warna pada 1 blok = panjang bit hash.`

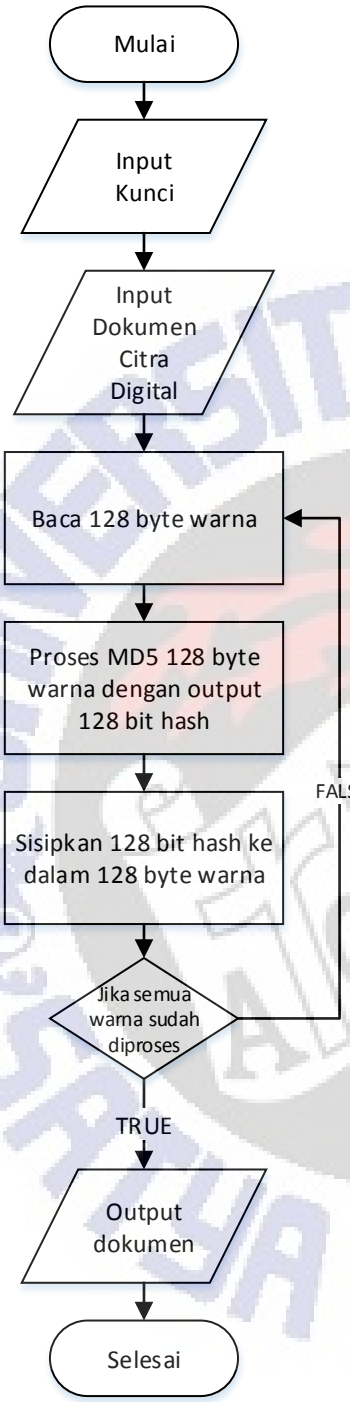
Jadi jika panjang *hash* adalah 64 *bit*, maka blok harus terdiri dari 64 warna, jika *hash* 128 *bit*, maka blok harus terdiri dari 128 warna. 1 warna akan memuat 1 *bit hash*.

Sebagai contoh, pada Gambar 5, jika digunakan algoritma *hash* yang menghasilkan nilai *hash* dengan panjang 12 *bit*, maka proses penyisipan

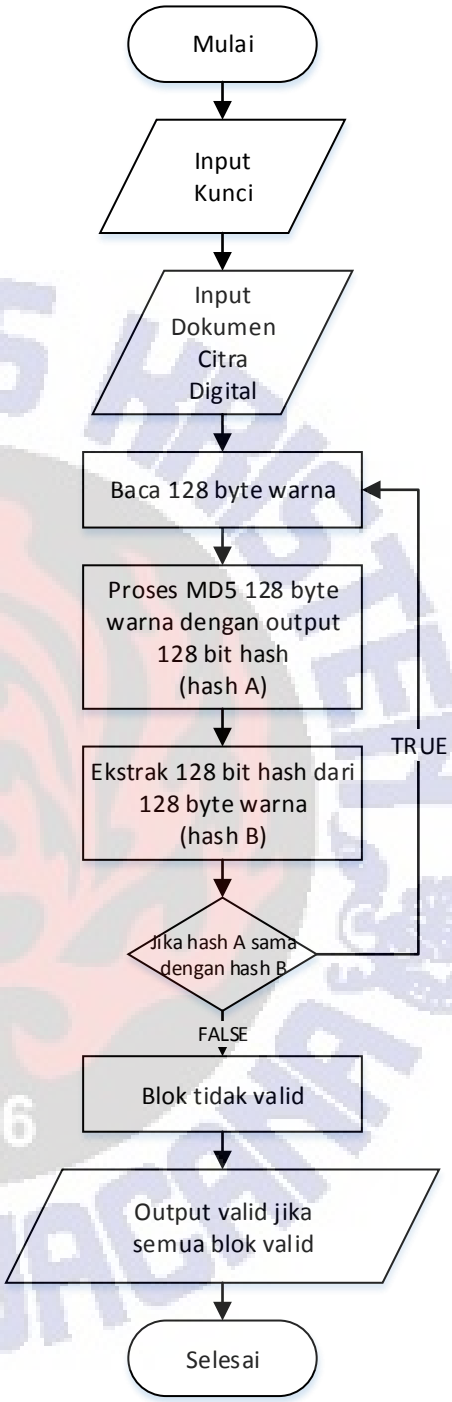
watermark ditunjukkan pada Tabel 1. Pada Tabel 1, ditambahkan proses enkripsi sebelum nilai *hash* disisipkan. Algorima enkripsi yang pada penelitian ini adalah Vigenere *cipher*.

Tabel 1 Contoh Proses *Block Based Watermark*

		Warna	*Hash*	Cipher *hash*	Hasil Penyisipan	
BLOK 1	200	11001000	1	1	11001001	201
	200	11001000	1	0	11001000	200
	100	01100100	0	0	01100100	100
	45	00101101	0	1	00101101	45
	55	00110111	1	1	00110111	55
	10	00001010	1	0	00001010	10
	90	01011010	1	1	01011011	91
	120	01111000	0	0	01111000	120
	90	01011010	1	0	01011010	90
	255	11111111	1	1	11111111	255
	0	00000000	1	0	00000000	0
	0	00000000	0	0	00000000	0
BLOK 2	80	01010000	0	1	01010001	81
	70	01000110	1	1	01000111	71
	0	00000000	1	0	00000000	0
	255	11111111	1	1	11111111	255
	255	11111111	0	0	11111110	254
	255	11111111	1	0	11111110	254
	90	01011010	1	0	01011010	90
	90	01011010	1	0	01011010	90
	10	00001010	1	0	00001010	10
	178	10110010	1	0	10110010	178
	190	10111110	1	0	10111110	190
	240	11110000	1	0	11110000	240
BLOK Selanjutnya						



Gambar 6 Alur Proses Pemberian *Watermark*



Gambar 7 Alur Proses Verifikasi *Watermark*

Proses pemberian *watermark* ditunjukkan pada Gambar 6. Proses ini memerlukan *input* dari pengguna yaitu kunci, dan dokumen citra *digital*. Kunci digunakan untuk menyandikan *hash*. *Hash* diperoleh dari proses algoritma MD5. *Hash* terenkripsi disisipkan ke dalam citra *digital*. Hasil akhir adalah dokumen citra *digital* yang telah diberi *watermark*.

Pada proses verifikasi (Gambar 7), *hash* yang telah disisipkan, diekstrak kemudian didekripsi. Hasil dekripsi dibandingkan dengan *hash* citra *digital* sekarang. Jika nilai *hash* ini berbeda, maka dapat dipastikan bahwa citra *digital* tersebut telah mengalami perubahan.

4. Hasil dan Pembahasan

Hasil implementasi sistem berdasarkan perancangan yang telah dibuat, dijelaskan sebagai berikut. Halaman pada program terbagi menjadi dua, yaitu *form* untuk proses *signing*, dan *form* untuk proses *verification*.

Gambar 8 *Form Signing*

Pada *form signing*, gambar yang akan diberi *watermark* dipilih. Kemudian user memasukkan kunci enkripsi. Gambar yang telah diberi *watermark*, ditampilkan disebelah gambar asli, dan kemudian user dapat menyimpan sebagai *file* baru.

Gambar 9 *Form Verification*

Pada proses verifikasi (Gambar 9), ditampilkan hasil akhir berupa valid atau tidak valid. Nilai *watermark* yang telah disisipkan sebelumnya dibandingkan dengan nilai *watermark* sekarang. Jika kunci yang digunakan untuk proses verifikasi berbeda dengan kunci pada proses pemberian *watermark*, maka nilai *watermark* akan memberikan hasil yang berbeda, sekalipun citra *digital* tidak mengalami perubahan (manipulasi). Sehingga hanya penerima yang sah, yang memiliki kunci yang tepat, yang dapat melakukan proses verifikasi.

Kode Program 1 Perintah untuk Membaca Warna pada Dokumen Gambar

```
1   public static byte[] ExtractColors(Bitmap img)
2   {
3           List<byte> list = new List<byte>();
4           Bitmap bitmap = img;
5
6           for (int y = 0; y < img.Height; y++)
7           {
8                   for (int x = 0; x < img.Width; x++)
9                   {
10                          Color c = bitmap.GetPixel(x, y);
11                          list.Add(c.R);
12                          list.Add(c.G);
13                          list.Add(c.B);
14                  }
15          }
16          return list.ToArray();
17  }
```

.Net Framework menyediakan *library* untuk mengolah dokumen dengan format PNG, yaitu dengan menggunakan *class* Bitmap. Melalui *class* ini dapat dilakukan proses manipulasi piksel yang ada di dalam suatu dokumen gambar.*Bit-*

bit watermark disisipkan pada LSB tiap warna pada piksel. Dalam satu piksel terdapat 3 warna yaitu RED, GREEN dan BLUE (baris 11-13). Dengan demikian dalam satu piksel dapat disisipi 3 *bit* data.

Kode Program 2 Perintah untuk Memanipulasi LSB

```
1   private static byte ReplaceLSB(byte current, char p)
2   {
3           string binary = Convert
4                   .ToString(current, 2)
5                   .PadLeft(8, '0');
6           char[] arrayBit = binary.ToCharArray();
7           arrayBit[7] = p;
8           binary = new string(arrayBit);
9           return Convert.ToByte(binary, 2);
10  }
```

Untuk mengubah LSB suatu *byte* warna, maka proses yang dilakukan adalah mengubah warna tersebut ke dalam format *binary*. Kemudian *bit* paling kanan dari warna tersebut diganti dengan *bit* pesan. Untuk mengubah *byte* menjadi *binary* digunakan *library class Convert* (baris 3).

Kode Program 3 Perintah untuk Membaca *Watermark*

```
11  public static byte[] GetLSB(byte[] data)
12  {
13          List<char> bits = new List<char>();
14          for (int i = 0; i < data.Length; i++)
15          {
16                  bits.Add(RetrieveLSB(data[i]));
17          }
18          int x = bits.Count % 8;
19          if (x != 0)
20          {
21                  for (int i = 0; i < x; i++)
22                  {
23                          bits.Add('0');
24                  }
25          }
26          byte[] message = GetBytesArray(
27          new string(bits.ToArray()));
28          return message;
29  }
```

Proses untuk membaca *watermark* terdari dari proses membaca warna pada dokumen gambar. Dilanjutkan dengan proses membaca nilai LSB dari tiap-tiap warna, dan dikumpulkan pada suatu variabel penampung (baris 3).

Pengujian sistem dilakukan terhadap beberapa faktor yaitu otentikasi, integritas. [9]. Otentikasi memiliki makna yaitu dokumen tersebut asli dan berasal dari sumber yang dipercaya [10]. Pengujian otentikasi dilakukan dengan menguji apakah dengan pasangan kunci yang berbeda, proses verifikasi dapat dilakukan. Tabel 2 menunjukkan hasil pengujian otentikasi, dan sistem dapat bekerja dengan tepat untuk mengetahui kunci yang digunakan benar atau tidak.

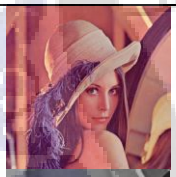
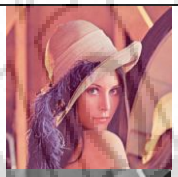
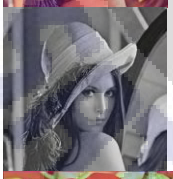
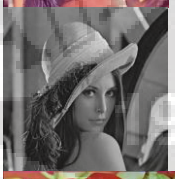
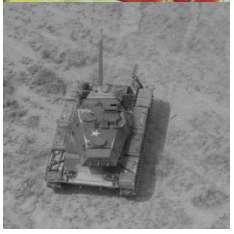
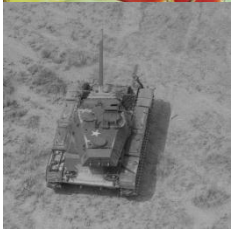
Tabel 2 Hasil Pengujian Otentikasi

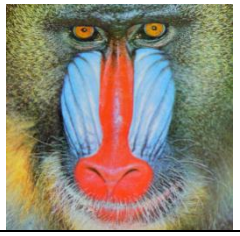
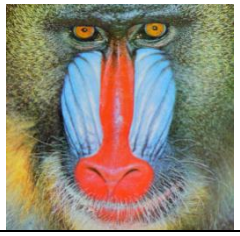
	Kunci *Signatory*	**Kunci*Verifier***	**Perbedaan karakter kunci**	***Output* proses verifikasi**	**Kesimpulan Pengujian**
1	123456	12345	1	Tidak otentik	Berhasil
2	123456	123457	1	Tidak otentik	Berhasil

3	ABC123rahasia	abC123rahasia	2	Tidak otentik	Berhasil
4	Abc123rahasia	aBc123rahasia	2	Tidak otentik	Berhasil
5	aBC123	aBC231	3	Tidak otentik	Berhasil
6	123456789	123456123	3	Tidak otentik	Berhasil
7	!@#Q@#!@#	!@#Q@3123	4	Tidak otentik	Berhasil
8	AAAAA1123	AAAAA!!@#	4	Tidak otentik	Berhasil
9	FTIFTIFTI	ftiFTIFti	5	Tidak otentik	Berhasil
10	ukswUKSW	ukswUKSW	5	Tidak otentik	Berhasil
11	123456	123456	0	Otentik	Berhasil
12	ABC123rahasia	ABC123rahasia	0	Otentik	Berhasil
13	Abc123rahasia	Abc123rahasia	0	Otentik	Berhasil
14	aBC123	aBC123	0	Otentik	Berhasil
15	123456789	123456789	0	Otentik	Berhasil

Watermark pada penelitian ini memiliki sifat *imperceptible* [11]. *Imperceptible* artinya adalah gambar semula dan gambar setelah proses watermarking, secara perceptual (secara visual) tidak terdapat perbedaan. Untuk menguji hal ini dilakukan langkah yaitu menggunakan 30 responen untuk dimintai pendapat. Tiap responden ditunjukkan 5 gambar tanpa *watermark*, dan 5 gambar yang sama yang telah diberi *watermark*, kemudian diberi pertanyaaan apakah ada perbedaan antara gambar awal dengan gambar dengan *watermark*.

Tabel 3 Hasil Pengujian Integritas

No	Citra Digital Awal	Citra Digital dengan *Watermark*	Jumlah Menjawab Sama	Jumlah Menjawab Tidak Sama	Kesimpulan Hasil Pengujian
1			30	0	Sukses
2			30	0	Sukses
3			30	0	Sukses
4			30	0	Sukses

5			30	0	Sukses

Berdasarkan hasil pengujian *imperceptible*, dapat disimpulkan bahwa proses watermarking, secara visual tidak menimbulkan perbedaan pada citra digital.

5. Simpulan

Berdasarkan penelitian, pengujian dan analisis terhadap aplikasi, maka dapat diambil kesimpulan yaitu *watermark* pada citra *digital* dapat diimplementasikan dengan cara yaitu menghitung nilai *hash* tiap blok-blok warna pada citra *digital*. Blok warna ini dibentuk dengan panjang sesuai dengan panjang *hash* dari algoritma yang digunakan. Pada penelitian ini digunakan MD5 yang menghasilkan blok sepanjang 128 *bit*. Tiap satu *bit hash*, disisipkan ke dalam 1 warna. Sebelum disisipkan, *hash* tersebut dienkripsi dengan algoritma Vigenere. Algoritma ini tidak mengubah panjang *cipher text* yang dihasilkan. Proses verifikasi dilakukan dengan cara membandingkan 128 *bit* yang disisipkan ini, dengan 128 *bit* hasil dari perhitungan *hash* pada saat verifikasi. Tiap blok akan memiliki nilai validitas sendiri-sendiri, sehingga dapat dideteksi lokasi kerusakan citra *digital* berdasarkan lokasi blok yang mengalami perubahan nilai *hash*. Berdasarkan hasil pengujian ditunjukkan bahwa watermark yang diimplementasikan pada penelitian ini telah memenuhi sifat imperceptible, yang diujikan dengan menggunakan bantuan 30 responden yang memberikan respon bahwa tidak ada perbedaan pada citra digital sebelum dan sesudah *watermarking*. Saran yang dapat diberikan untuk penelitian lebih lanjut adalah perlu dilakukan uji perbandingan dengan algoritma *hash* yang lain. Algoritma *hash* yang ada akan memberikan keuntungan dan kerugian. Hal ini perlu diketahui sehingga dapat ditentukan algoritma *hash* yang ideal, berdasarkan bentuk dan ukuran citra *digital*.

6. Daftar Pustaka

[1]. Rivest, R. L. 1992. *RFC 1321: The MD5 message-digest algorithm*. Internet activities board 143.

[2]. Gaitonde, S. S. & Ramabadran, T. V 1988. *A tutorial on CRC computations*. IEEE Micro 8, 62–75.

[3]. Setiawan, R. B. & Made, I. 2012. *Perancangan Aplikasi Watermarking pada Media Fotografi Sebagai Perlindungan Hak Cipta Menggunakan Metode Spread Spectrum*.

[4]. Ady, P. 2008. *Teknik Pembuktian Kepemilikan Citra Digital dengan Watermarking pada Domain Wavelet*. Prosiding SP MIPA

[5]. Johnson, N. F., Duric, Z. & Jajodia, S. 2001. *Information hiding:*

steganography and watermarking: attacks and countermeasures. Springer.
[6]. Amin, M. M. 2015. *Image Steganography dengan Metode Least Significant Bit (LSB)*. CSRID Journal 6, 53–64.
[7]. Wong, P. W. & Memon, N. 2001. *Secret and public key image watermarking schemes for image authentication and ownership verification*. Image Processing, IEEE Transactions on 10, 1593–1601.
[8]. Walia, A. G. N. K. 2014. *Cryptography Algorithms: A Review*. International Journal of Engineering Development and Research
[9]. Shaw, S. 1999. *Overview of Watermarks , Fingerprints , and Digital Signatures*.
[10]. Stallings, W. 2006. *Cryptography and Network Security*. (doi:10.1007/11935070)
[11]. Kaur, M., Jindal, S. & Behal, S. 2012. *A study of digital image watermarking*. Journal of Research in Engineering and Applied Sciences 2, 126–136.